

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS
GERAIS

ENGENHARIA DE COMPUTAÇÃO

Apresentação da Solução de Banco de Dados

PROJETO FÍSICO - IMPLEMENTAÇÃO (SGBD)

Bárbara Maria Sampaio Portes
Samir da Morim Cambraia

Belo Horizonte, 30 de junho de 2025

Projeto Físico - Implementação do Banco de Dados

Abaixo está o script SQL completo para criar o banco de dados **PetFeeder** em um SGBD como MySQL/MariaDB, incluindo tabelas, relacionamentos, índices e restrições de integridade.

1. Criação do Banco de Dados

```
1 CREATE DATABASE IF NOT EXISTS PetFeeder;  
2 USE PetFeeder;
```

2. Criação das Tabelas

2.1. Tabela DISPOSITIVO

```
1 CREATE TABLE IF NOT EXISTS DISPOSITIVO (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     nome VARCHAR(100) NOT NULL,  
4     id_mqtt VARCHAR(50) NOT NULL UNIQUE  
5 ) ENGINE=InnoDB;
```

2.2. Tabela SENSOR

```
1 CREATE TABLE IF NOT EXISTS SENSOR (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     id_dispositivo INT NOT NULL,  
4     tipo VARCHAR(50) NOT NULL,  
5     topico_alerta VARCHAR(100) NOT NULL,  
6     FOREIGN KEY (id_dispositivo) REFERENCES DISPOSITIVO(id) ON DELETE  
    CASCADE  
7 ) ENGINE=InnoDB;
```

2.3. Tabela LEITURA

```
1 CREATE TABLE IF NOT EXISTS LEITURA (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     id_sensor INT NOT NULL,  
4     valor VARCHAR(50) NOT NULL,  
5     data_hora DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,  
6     FOREIGN KEY (id_sensor) REFERENCES SENSOR(id) ON DELETE CASCADE  
7 ) ENGINE=InnoDB;
```

2.4. Tabela ATUADOR

```
1 CREATE TABLE IF NOT EXISTS ATUADOR (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     id_dispositivo INT NOT NULL,  
4     tipo VARCHAR(50) NOT NULL,  
5     topico_comando VARCHAR(100) NOT NULL,  
6     FOREIGN KEY (id_dispositivo) REFERENCES DISPOSITIVO(id) ON DELETE  
    CASCADE  
7 ) ENGINE=InnoDB;
```

2.5. Tabela COMANDO

```
1 CREATE TABLE IF NOT EXISTS COMANDO (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     id_atuador INT NOT NULL,  
4     comando VARCHAR(50) NOT NULL,  
5     data_hora DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,  
6     FOREIGN KEY (id_atuador) REFERENCES ATUADOR(id) ON DELETE CASCADE  
7 ) ENGINE=InnoDB;
```

3. Índices para Melhorar Desempenho

```
1 -- Índice para buscas por tipo de sensor  
2 CREATE INDEX idx_sensor_tipo ON SENSOR(tipo);  
3  
4 -- Índice para buscas por tipo de atuador  
5 CREATE INDEX idx_atuador_tipo ON ATUADOR(tipo);  
6  
7 -- Índice para consultas de leituras por sensor e data  
8 CREATE INDEX idx_leitura_sensor_data ON LEITURA(id_sensor, data_hora);  
9  
10 -- Índice para consultas de comandos por atuador e data  
11 CREATE INDEX idx_comando_atuador_data ON COMANDO(id_atuador, data_hora);
```

4. Inserção de Dados de Exemplo (Opcional)

```
1 -- Inserir um dispositivo de exemplo  
2 INSERT INTO DISPOSITIVO (nome, id_mqtt)  
3 VALUES ('PetFeeder Casa', 'MQTT-PET_001');  
4  
5 -- Inserir um sensor associado ao dispositivo  
6 INSERT INTO SENSOR (id_dispositivo, tipo, topico_alerta)  
7 VALUES (1, 'Nivel de Racao', 'petfeeder/001/sensor/nivel');  
8  
9 -- Inserir uma leitura do sensor  
10 INSERT INTO LEITURA (id_sensor, valor)  
11 VALUES (1, '75.5');  
12  
13 -- Inserir um atuador associado ao dispositivo  
14 INSERT INTO ATUADOR (id_dispositivo, tipo, topico_comando)  
15 VALUES (1, 'Dispensador de Racao', 'petfeeder/001/atuador/dispensar');  
16  
17 -- Inserir um comando para o atuador  
18 INSERT INTO COMANDO (id_atuador, comando)  
19 VALUES (1, 'ON');
```

5. Consultas Úteis para o Sistema

5.1. Buscar todos os sensores de um dispositivo

```
1 SELECT s.id, s.tipo, s.topico_alerta  
2 FROM SENSOR s  
3 WHERE s.id_dispositivo = 1;
```

5.2. Obter as últimas 10 leituras de um sensor

```
1 SELECT l.valor, l.data_hora
2 FROM LEITURA l
3 WHERE l.id_sensor = 1
4 ORDER BY l.data_hora DESC
5 LIMIT 10;
```

5.3. Listar todos os atuadores e seus últimos comandos

```
1 SELECT
2     a.tipo,
3     a.topico_comando,
4     c.comando,
5     c.data_hora
6 FROM ATUADOR a
7 LEFT JOIN COMANDO c ON a.id = c.id_atuador
8 WHERE a.id_dispositivo = 1
9 ORDER BY c.data_hora DESC;
```

5.4. Verificar dispositivos com sensores de nível baixo

```
1 -- Exemplo para valores numericos. Adaptar se 'valor' for texto.
2 SELECT
3     d.nome,
4     s.tipo,
5     l.valor,
6     l.data_hora
7 FROM DISPOSITIVO d
8 JOIN SENSOR s ON d.id = s.id_dispositivo
9 JOIN LEITURA l ON s.id = l.id_sensor
10 WHERE s.tipo = 'Nivel de Racao' AND CAST(l.valor AS DECIMAL) < 20
11 ORDER BY l.data_hora DESC;
```